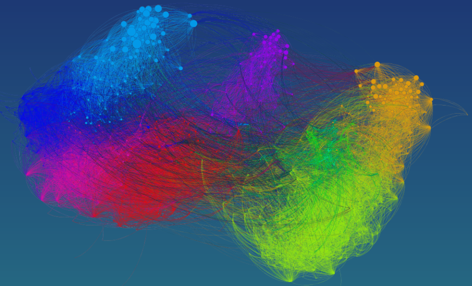
Graphs in Machine Learning

Michal Valko

Inria Lille - Nord Europe, France

TA: Pierre Perrault

Partially based on material by: Andreas Krause, Branislav Kveton, Michael Kearns



Piazza for Q&A's

Purpose

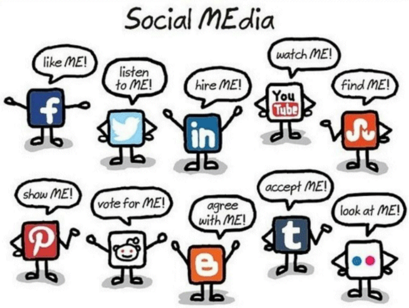
- registration for the class
- register with your **school** email and **full name**
- online course discussions and announcements
- questions and answers about the material and logistics
- **students encouraged to answer each others' questions**
- homework assignments
- virtual machine link and instructions
- **draft of the slides before the class**

https://piazza.com/ens_cachan/fall2018/mvagraphsml **NO EMAILS!**

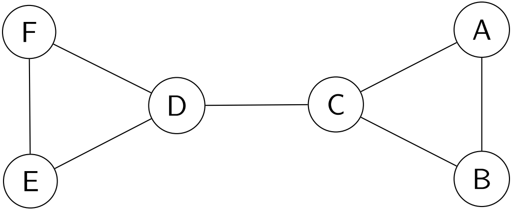
class code given during the class

Graphs from social networks

- people and their interactions
- directed (Twitter) and undirected (Facebook)
- structure is rather a *phenomena*
- typical ML tasks
  - advertising
  - product placement
  - link prediction (PYMK)

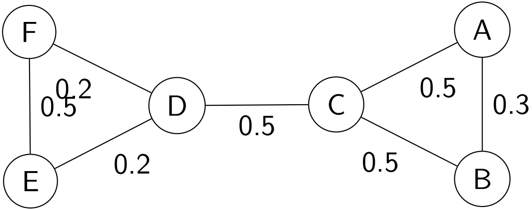


Success story #1 Product placement - problem



Maximizing the Spread of Influence through a Social Network
http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf

Success story #1 Product placement - problem

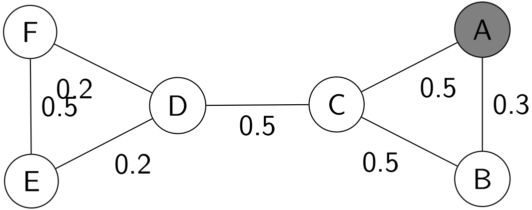


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 Product placement - problem

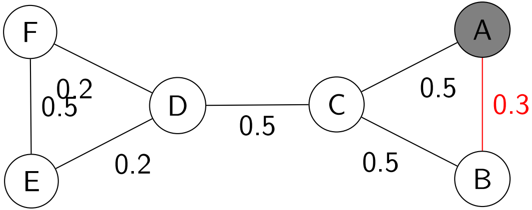


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 Product placement - problem

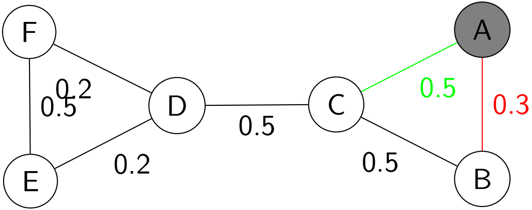


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 Product placement - problem

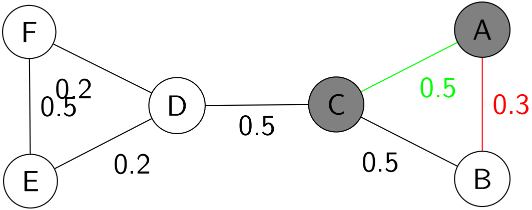


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 Product placement - problem

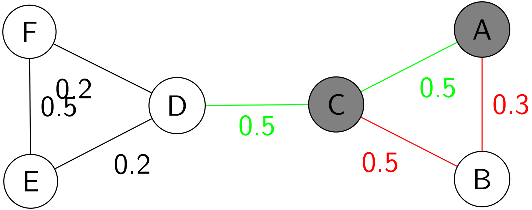


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf

Success story #1 Product placement - problem

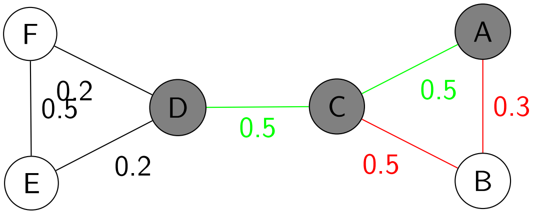


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf

Success story #1 Product placement - problem

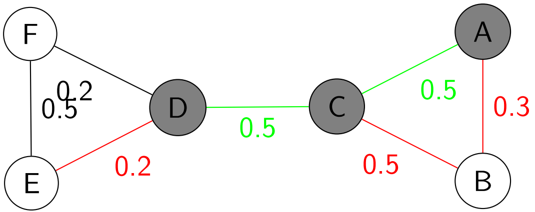


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 Product placement - problem

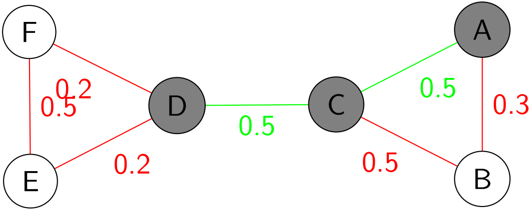


Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Success story #1 Product placement - problem



Who should get free cell phones?

$V = \{$**A**lice,**B**ob,**C**harlie,**D**orothy,**E**ric,**F**iona$\}$

$F(S) =$ Expected number of people influenced when targeting $S \subseteq V$ under some propagation model - e.g., cascades

How would you choose the target customers?

highest degree, close to the center, . . .

Maximizing the Spread of Influence through a Social Network
`http://www.cs.cornell.edu/home/kleinber/kdd03-inf.pdf`

Submodularity: Definition

A **set function** on a discrete set A is **submodular** if for any $S \subseteq T \subseteq A$ and for any $e \in A \setminus T$

$$f(S \cup \{e\}) - f(S) \geq f(T \cup \{e\}) - f(T)$$

Example: $S = \{\text{stuff}\} = \{\text{bread, apple, tomato, ...}\}$

$f(V) =$ cost of getting products V

$f(\{\text{bread}\}) = c(\text{bakery}) + c(\text{bread})$

$f(\{\text{bread}, \text{apple}\}) = c(\text{bakery}) + c(\text{bread}) + c(\text{market}) + c(\text{apple}\})$

$f(\{\text{bread}, \text{tomato}\}) = c(\text{bakery}) + c(\text{bread}) + c(\text{market}) + c(\text{tomato})$

$f(\{\text{bread}, \text{tomato}, \text{apple}\}) = c(\text{bakery}) + c(\text{bread}) + c(\text{market}) + c(\text{tomato}) + c(\text{apple})$

Adding an apple to the smaller set costs more!

$$\{\text{bread}\} \subseteq \{\text{bread}, \text{tomato}\}$$

$$f(\{\text{bread}, \text{apple}\}) - f(\{\text{bread}\}) > f(\{\text{bread}, \text{tomato}, \text{apple}\}) - f(\{\text{tomato}, \text{bread}\})$$

Diminishing returns: Buying in bulk is cheaper!

Submodularity: Application

Objective: Find $\arg\max_{S \subseteq A, |S| \leq k} f(S)$

Property: NP-hard in general

Special case: f is also **nonnegative** and **monotone**.

Other examples: information, graph cuts, covering, ...

Link to our **product placement** problem on a **social network graph?**

submodular?, nonnegative?, monotone?, k?

http://thibaut.horel.org/submodularity/papers/nemhauser1978.pdf

Let $S^\star = \arg\max_{S \subseteq A, |S| \leq k} f(S)$ where f is monotonic and submodular set function and let $S_{\mathtt{Greedy}}$ be a **greedy solution**.

$$\text{Then} \quad f(S_{\mathtt{Greedy}}) \geq \left(1 - \frac{1}{e}\right) \cdot f(S^\star).$$

Submodularity: Greedy algorithm

Input:
  k: the maximum allowed cardinality of the output
  V: a ground set
  f: a monotone, non-negative, and submodular function
Run:
$S_0 = \emptyset$
for $i = 1$ **to** k **do**
  $S_i \leftarrow S_{i-1} \cup \left\{ \arg\max_{a \in V \setminus S_{i-1}} \left[f\left(\{a\} \cup S_{i-1}\right) - f\left(S_{i-1}\right) \right] \right\}$
end for
Output:
  Return $S_{\texttt{Greedy}} = S_k$

Let $S^\star = \arg\max_{S \subseteq A, |S| \leq k} f(S)$ where f is monotonic and submodular set function and let $S_{\texttt{Greedy}}$ be a **greedy solution**.

$$\text{Then} \quad f(S_{\texttt{Greedy}}) \geq \left(1 - \frac{1}{e}\right) \cdot f(S^\star).$$

Submodularity: Approximation guarantee of Greedy

Let S_i be the i-th set selected by Greedy, $S_{\texttt{Greedy}} = S_k$. We show

$$f(S^\star) - f(S_i) \leq \left(1 - \frac{1}{k}\right)^i \cdot f(S^\star).$$

Difference from the optimum before the i-th step ...

$$\begin{aligned}
f\left(S^\star\right) - f\left(S_{i-1}\right) &\leq f\left(S^\star \cup S_{i-1}\right) - f\left(S_{i-1}\right) \\
&\leq \sum_{a \in S^\star \setminus S_{i-1}} \left(f\left(\{a\} \cup S_{i-1}\right) - f\left(S_{i-1}\right)\right) \\
&\leq \sum_{a \in S^\star \setminus S_{i-1}} \left(f\left(S_i\right) - f\left(S_{i-1}\right)\right) \\
&\leq k\left(f\left(S_i\right) - f\left(S_{i-1}\right)\right)
\end{aligned}$$

Difference from the optimum after the i-th step ...

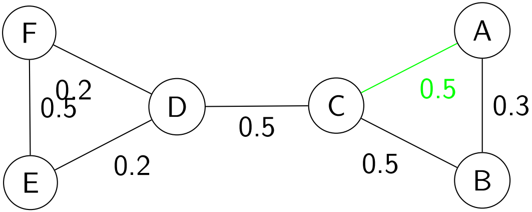
$$\begin{aligned}
f(S^\star) - f(S_i) &= f(S^\star) - f(S_{i-1}) - \left(f\left(S_i\right) - f\left(S_{i-1}\right)\right) \\
&\leq f(S^\star) - f(S_{i-1}) - \frac{f(S^\star) - f(S_{i-1})}{k}
\end{aligned}$$

Submodularity: Graph-related examples

- Influence maximization on networks (current example)
- Maximum-weight spanning trees
- Graph cuts
- Structure learning in graphical models (PGM course)
- More examples http://people.math.gatech.edu/~tetali/LINKS/IWATA/SFGT.pdf
- Deep Submodular Functions (2017) https://arxiv.org/pdf/1701.08939.pdf

back to the influence-maximization example ...

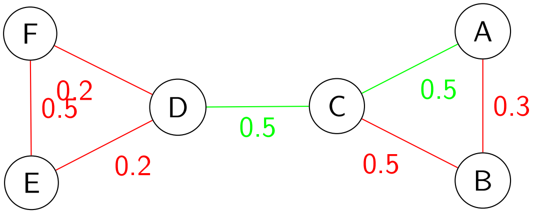
Success story #1 Product placement - solution



Key idea: Flip coins c in advance → "live" edges

MIIA: http://hanj.cs.illinois.edu/pdf/dmkd12_cwang.pdf/
Tutorial: cf. Andreas Krause http://submodularity.org/
Course: Jeff Billmes at UW

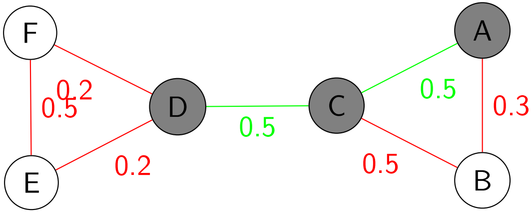
Success story #1 Product placement - solution



Key idea: Flip coins c in advance → “live” edges

MIIA: http://hanj.cs.illinois.edu/pdf/dmkd12_cwang.pdf/
Tutorial: cf. Andreas Krause http://submodularity.org/
Course: Jeff Billmes at UW

Success story #1 Product placement - solution

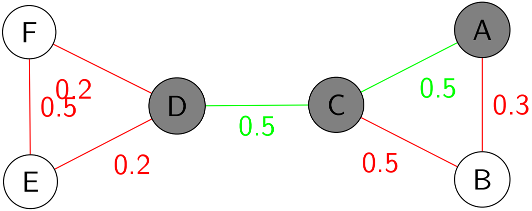


Key idea: Flip coins c in advance $\rightarrow$ “live” edges

$F_c(V)$ = People influenced under outcome c (set cover!)

MIIA: http://hanj.cs.illinois.edu/pdf/dmkd12_cwang.pdf/
Tutorial: cf. Andreas Krause http://submodularity.org/
Course: Jeff Billmes at UW

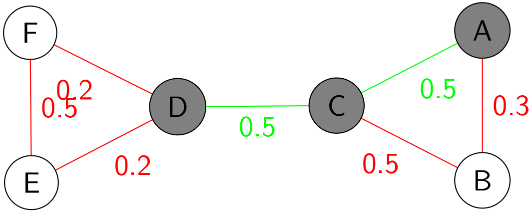
Success story #1 Product placement - solution



Key idea: Flip coins c in advance → "live" edges
$F_c(V)$ = People influenced under outcome c (set cover!)
$F(V) = \sum_c P(c)F_c(V)$ is submodular as well!

MIIA: http://hanj.cs.illinois.edu/pdf/dmkd12_cwang.pdf/
Tutorial: cf. Andreas Krause http://submodularity.org/
Course: Jeff Billmes at UW

Success story #1 Product placement - solution



Key idea: Flip coins c in advance $\rightarrow$ "live" edges

$F_c(V)$ = People influenced under outcome c (set cover!)

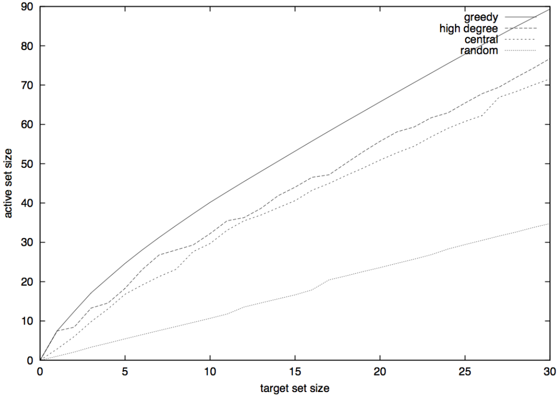
$F(V) = \sum_c P(c)F_c(V)$ is submodular as well!

Computational issues?

MIIA: http://hanj.cs.illinois.edu/pdf/dmkd12_cwang.pdf/
Tutorial: cf. Andreas Krause http://submodularity.org/
Course: Jeff Billmes at UW

Success story #1 Product placement - comparison

influence on the ArXiv/Physics co-authorship graph



greedy approximation does better than the centrality measures

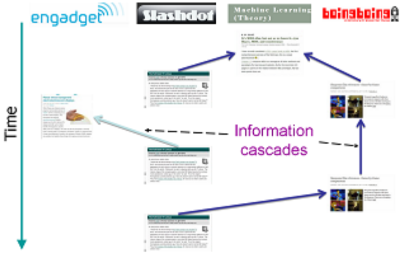
Graphs from utility and technology networks

- link services
- power grids, roads, transportation networks, Internet, sensor networks, water distribution networks
- structure is either *hand designed* or not
- typical ML tasks
  - best routing under unknown or variable costs
  - identify the node of interest

Berkeley's Floating Sensor Network

Graphs from information networks

- web
- blogs
- wikipedia
- typical ML tasks
  - find influential sources
  - search (PageRank)



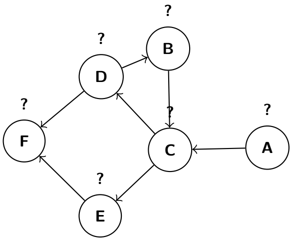
Blog cascades (ETH) - *submodularity*

Success story #2 Google PageRank

Objective: **Rank** all web pages (nodes on the graph) by how **many** other pages link to them and how **important** they are.

basic PageRank is independent of query and the page content

Internet → graph → matrix → stochastic matrix $\mathbf{M}$ $\left(\sum_j \mathbf{M}_{ij} = 1\right)$



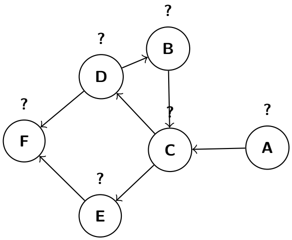
Success story #2 Goog… …eRank

Objective: **Rank** … …ph) by how **many** other pages lin… …e.

basic PageRan… …ge content

Internet → graph … …trix **M** $\left(\sum_j \mathbf{M}_{ij} = 1\right)$

Random Surfer Process



Success story #2 Goog eRank

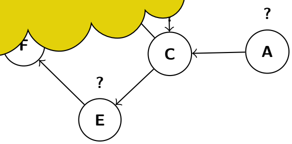
Objective: **Rank** ph) by how **many**
other pages lir e.

basic PageRan e content

Internet → graph trix $\mathbf{M}$ $\left(\sum_j \mathbf{M}_{ij} = 1\right)$

Random Surfer Process

What is wrong with it?



Success story #2 Google PageRank

http://infolab.stanford.edu/~backrub/google.html:

PageRank can be thought of as a model of user behavior. We assume there is a "random surfer" who is given a web page at random and keeps clicking on links, never hitting "back" but eventually gets bored and starts on another random page.

- page is **important** if **important** pages link **to** it
  - circular definition
- importance of a page is distributed **evenly**
- probability of being bored is 15%

Success story #2 Google PageRank

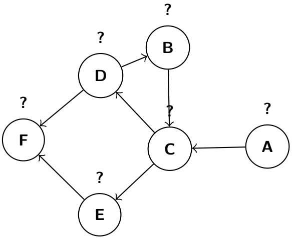
Google matrix: $\mathbf{G} = (1-p)\mathbf{M} + p \cdot \frac{1}{N}\mathbb{1}_{N\times N}$, where $p = 0.15$

Success story #2 Google PageRank

Google matrix: $\mathbf{G} = (1-p)\mathbf{M} + p \cdot \frac{1}{N}\mathbb{1}_{N\times N}$, where $p = 0.15$

$\mathbf{G}$ is **stochastic** why? What is $\mathbf{Ga}$ for any $\mathbf{a}$? We look for $\mathbf{Gv} = 1 \times \mathbf{v}$, steady-state vector, a right eigenvector with eigenvalue 1. why?

Perron's theorem: Such v exists and it is **unique**
if the entries of $\mathbf{G}$ are positive.

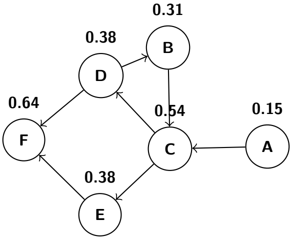


Success story #2 Google PageRank

Google matrix: $\mathbf{G} = (1-p)\mathbf{M} + p \cdot \frac{1}{N}\mathbb{1}_{N\times N}$, where $p = 0.15$

G is **stochastic** why? What is **Ga** for any **a**? We look for $\mathbf{Gv} = 1 \times \mathbf{v}$, steady-state vector, a right eigenvector with eigenvalue 1. why?

Perron's theorem: Such v exists and it is **unique**
if the entries of **G** are positive.

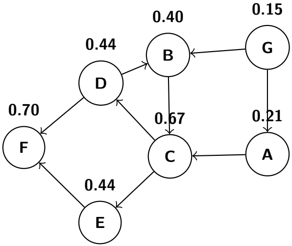


Success story #2 Google PageRank

Google matrix: $\mathbf{G} = (1-p)\mathbf{M} + p \cdot \frac{1}{N}\mathbb{1}_{N\times N}$, where $p = 0.15$

G is **stochastic** why? What is **Ga** for any **a**? We look for $\mathbf{Gv} = 1 \times \mathbf{v}$, steady-state vector, a right eigenvector with eigenvalue 1. why?

Perron's theorem: Such v exists and it is **unique**
if the entries of **G** are positive.



Success story #2 Google PageRank

History: [Desikan, 2006]

- The anatomy of a large-scale hypertextual web search engine [Brin & Page 1998]
- US patent for PageRank granted in 2001
- Google indexes 10's of billions of web pages (1 billion $= 10^9$)
- Google serves $\geq$ 200 million queries per day
- Each query processed by $\geq$ 1000 machines
- All search engines combined process more than 500 million queries per day

Success story #2 Google PageRank

Problem: Find an eigenvector of a stochastic matrix.

- $n = 10^9$!!!
- luckily: **sparse** (average outdegree: 7)
- better than a simple centrality measure (e.g., degree)
- power method

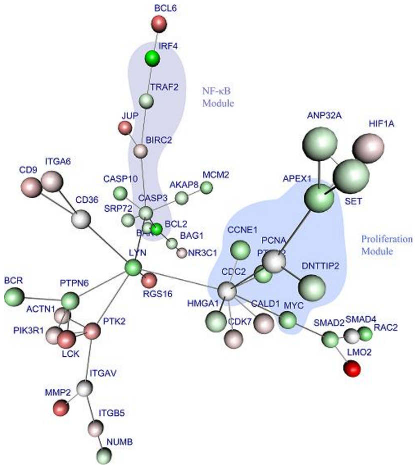
$$\begin{aligned} \mathbf{v}_0 &= (1_A \quad 0_B \quad 0_C \quad 0_D \quad 0_E \quad 0_F)^\mathsf{T} \\ \mathbf{v}_1 &= \mathbf{G}\mathbf{v}_0 \\ \mathbf{v}_{t+1} &= \mathbf{G}\mathbf{v}_t = \mathbf{G}^{t+1}\mathbf{v} \end{aligned}$$

$$\mathbf{v}_{t+1} = \mathbf{v}_t \implies \mathbf{G}\mathbf{v}_t = \mathbf{v}_t$$ and we found the steady vector

But wait, **M** is sparse, but **G** is dense! What to do?

Graphs from biological networks

- protein-protein interactions
- gene regulatory networks
- typical ML tasks
  - discover unexplored interactions
  - learn or reconstruct the structure



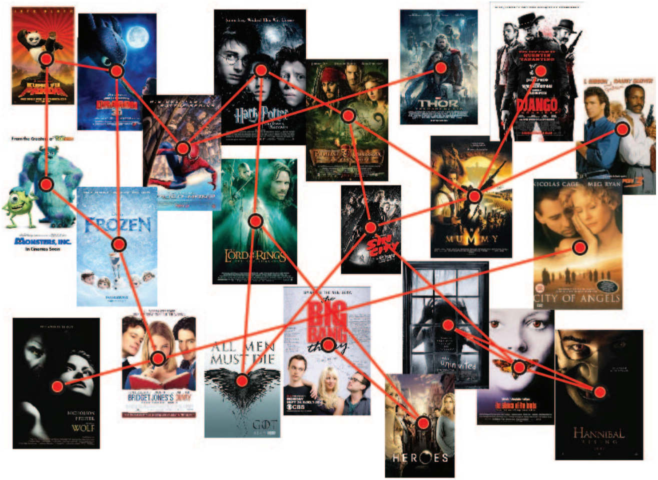
Diffuse large B-cell lymphomas - Dittrich et al. (2008)

Graphs from similarity networks

graph is not naturally given

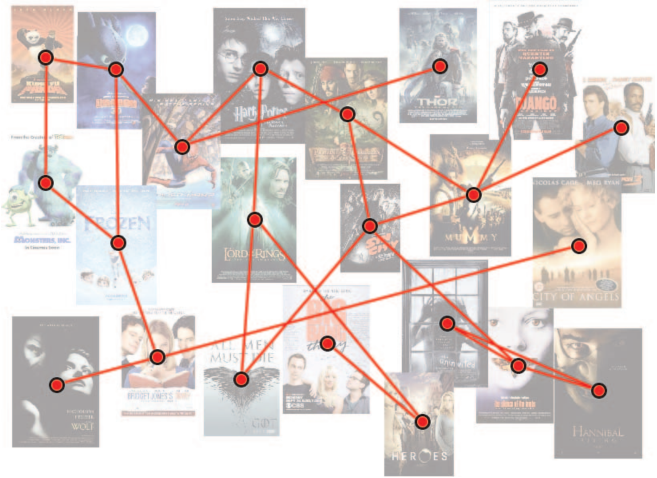
Graphs from similarity networks

but we can construct it



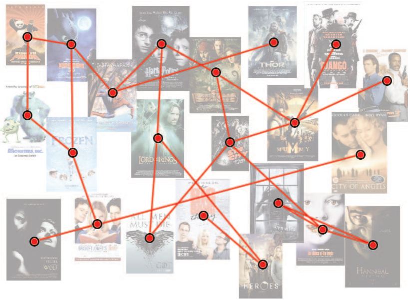
Graphs from similarity networks

and use it as an abstraction



Graphs from similarity networks

- vision
- audio
- text
- typical ML tasks
  - **semi-supervised learning**
  - **spectral clustering**
  - **manifold learning**



movie similarity

Two sources of graphs in ML

Graph as models for networks

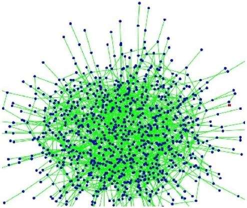
- given as an input
- discover interesting properties of the structure
- represent useful information (viral marketing)
- be the object of study (anomaly detection)

Graph as nonparametric basis

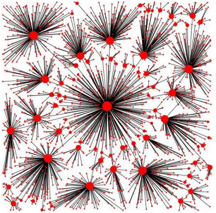
- we create (learn) the structure
- flat vectorial data $\rightarrow$ similarity graph
- nonparametric regularizer
- encode structural properties: smoothness, independence, ...

Random Graph Models

Erdős-Rényi
independent edges



Barabási-Albert
preferential attachment



Stochastic Blocks
modeling communities

Watts-Strogatz, Chung-Lu, Fiedler, ...

What will you learn in the Graphs in ML course?

Concepts, tools, and methods to work with graphs in ML.

Theoretical toolbox to analyze graph-based algorithms.

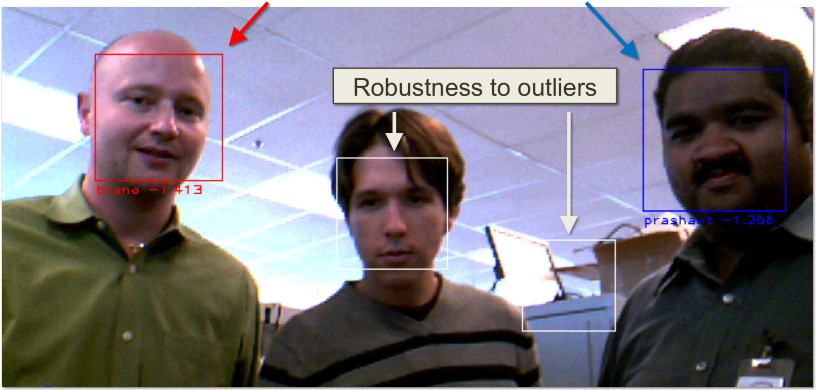
Specific applications of graphs in ML.

How to tackle: *large graphs, online setting, graph construction* ...

One example: Online Semi-Supervised Face Recognition

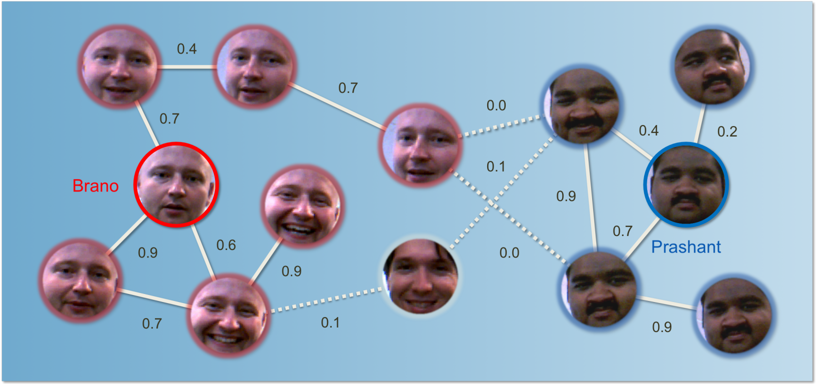
Online Semi-Supervised Face Recognition

graph is not given



Online Semi-Supervised Face Recognition

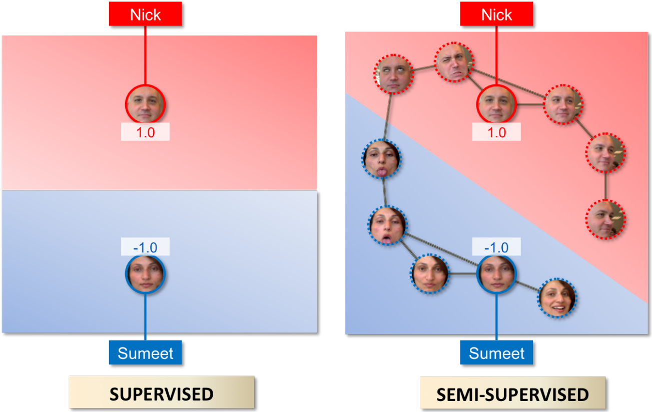
we will construct it!



An example of a similarity graph over faces. The faces are vertices of the graph. The edges of the graph connect similar faces. Labeled faces are outlined by thick solid lines.

Online Semi-Supervised Face Recognition

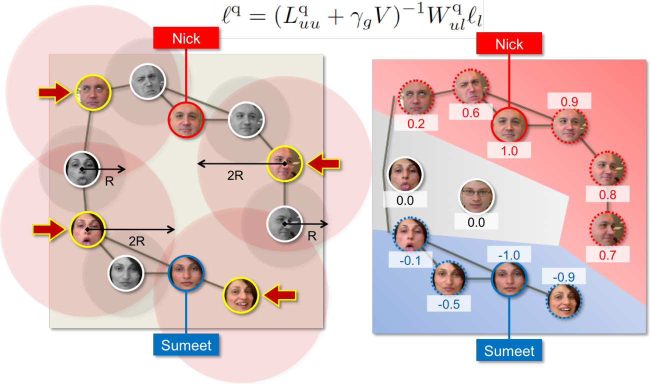
graph-based semi-supervised learning



Online Semi-Supervised Face Recognition

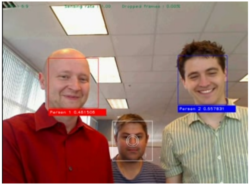
online learning - graph sparsification

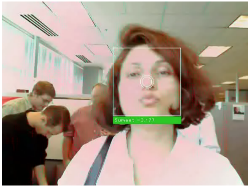
$$\ell^{q} = (L^{q}_{uu} + \gamma_g V)^{-1} W^{q}_{ul} \ell_l$$



DEMO

second TD





see the demo: `http://researchers.lille.inria.fr/~valko/hp/serve.php?what=publications/kveton2009nipsdemo.officespace.mov`

OSS FaceReco: Analysis

$$\frac{1}{n}\sum_t(\ell_t^{\mathrm{q}}[t]-y_t)^2 \le \frac{3}{n}\sum_t(\ell_t^*-y_t)^2 + \frac{3}{n}\sum_t(\ell_t^{\mathrm{o}}[t]-\ell_t^*)^2 + \frac{3}{n}\sum_t(\ell_t^{\mathrm{q}}[t]-\ell_t^{\mathrm{o}}[t])^2$$

Error of our solution	Offline learning error	Online learning error	Quantization error

Claim: When the regularization parameter is set as $\gamma_g = \Omega(n_l^{3/2})$, the difference between the risks on labeled and all vertices decreases at the rate of $O(n_l^{-1/2})$ (with a high probability)

$$\frac{1}{n}\sum_t(\ell_t^*-y_t)^2 \le \frac{1}{n_l}\sum_{i\in l}(\ell_i^*-y_i)^2 + \beta + \sqrt{\frac{2\ln(2/\delta)}{n_l}}(n_l\beta+4)$$

$$\beta \le \left[\frac{\sqrt{2}}{\gamma_g+1} + \sqrt{2n_l}\frac{1-\sqrt{c_u}}{\sqrt{c_u}}\frac{\lambda_M(L)+\gamma_g}{\gamma_g^2+1}\right]$$

OSS FaceReco: Analysis

$$\frac{1}{n}\sum_t (\ell_t^q[t]-y_t)^2 \le \frac{3}{n}\sum_t (\ell_t^* - y_t)^2 + \frac{3}{n}\sum_t (\ell_t^o[t]-\ell_t^*)^2 + \frac{3}{n}\sum_t (\ell_t^q[t]-\ell_t^o[t])^2$$

Error of our solution	Offline learning error	Online learning error	Quantization error

Claim: When the regularization parameter is set as $\gamma_g = \Omega(n^{1/4})$, the average error between the offline and online HFS predictions decreases at the rate of $O(n^{-1/2})$

$$\frac{1}{n}\sum_t (\ell_t^o[t]-\ell_t^*)^2 \le \frac{1}{n}\sum_t \left\|\ell^o[t]-\ell^*\right\|_2^2 \le \frac{4n_l}{(\gamma_g+1)^2}$$

$$\|\ell\|_2 \le \frac{\|y\|_2}{\lambda_m(C^{-1}K+I)} = \frac{\|y\|_2}{\lambda_m(K)\lambda_M^{-1}(C)+1} \le \frac{\sqrt{n_l}}{\gamma_g+1}$$

OSS FaceReco: Analysis

$$\frac{1}{n}\sum_t (\ell_t^{q}[t]-y_t)^2 \le \frac{3}{n}\sum_t (\ell_t^{*}-y_t)^2 + \frac{3}{n}\sum_t (\ell_t^{o}[t]-\ell_t^{*})^2 + \frac{3}{n}\sum_t (\ell_t^{q}[t]-\ell_t^{o}[t])^2$$

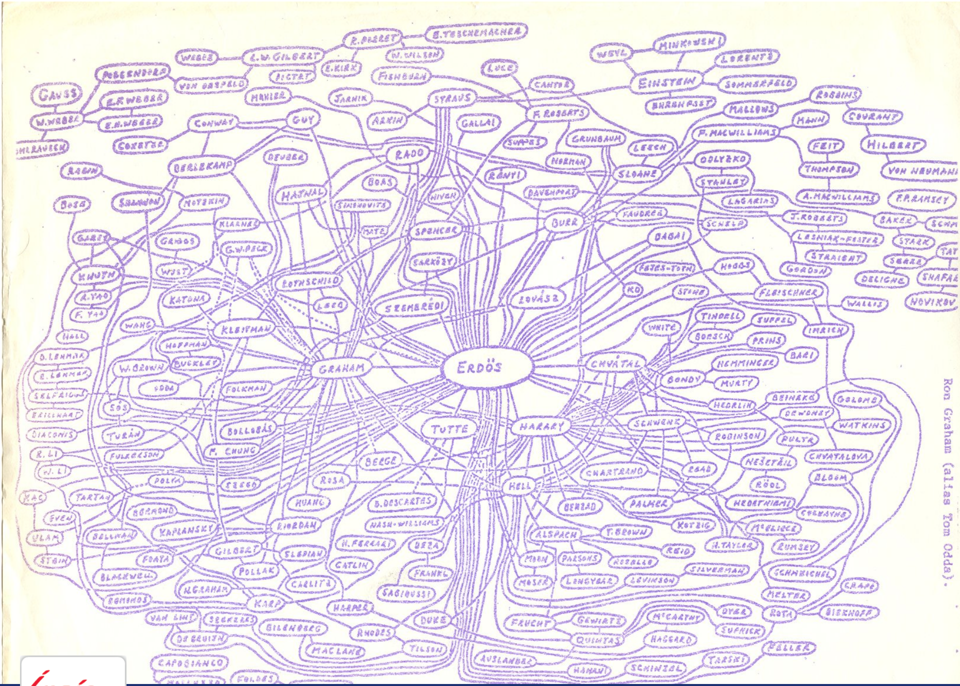
Error of our solution	Offline learning error	Online learning error	Quantization error

Claim: When the regularization parameter is set as $\gamma_g = \Omega(n^{1/8})$, and the Laplacians L^q and L^o and normalized, the average error between the online and online quantized HFS predictions decreases at the rate of $O(n^{-1/2})$

$$\frac{1}{n}\sum_t (\ell_t^{q}[t]-\ell_t^{o}[t])^2 \le \frac{1}{n}\sum_t \left\|\ell^{q}[t]-\ell^{o}[t]\right\|_2^2 \le \frac{n_l}{c_u^2\gamma_g^4}\left\|L^q - L^o\right\|_F^2$$

$$\left\|L^q - L^o\right\|_F^2 \propto O(k^{-2/d})$$

The distortion rate of online k-center clustering is $O(k^{-1/d})$, where d is dimension of the manifold and k is the number of representative vertices

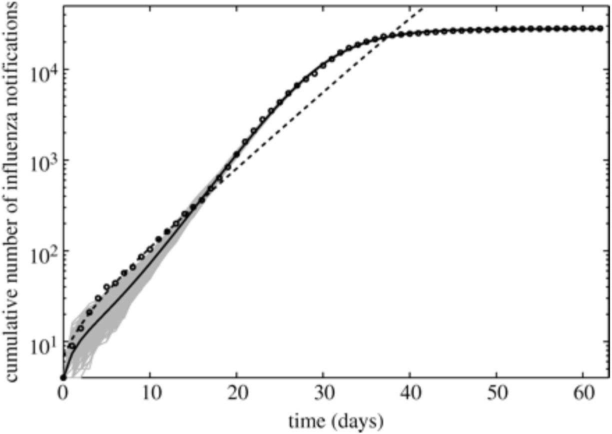

ERDÖS
GRAHAM
TUTTE
HARARY
EINSTEIN
MINKOWSKI
WEYL
LORENTZ
SOMMERFELD
GAUSS
W.WEBER
CONWAY
GUY
STRAUS
BERLEKAMP
RADO
SZEMERÉDI
LOVÁSZ
CHVÁTAL
KLEITMAN
F. CHUNG
BOLLOBÁS
HILBERT
VON NEUMANN
ODLYZKO
SLOANE
KNUTH
BERGE
HELL
ALSPACH
DUKE
ULAM
Ron Graham (alias Tom Odda).

Erdős number project

- http://www.oakland.edu/enp/ **try it!**
- an example of a real-world graph
- 401 000 authors, 676 000 edges ($\ll 401000^2 \rightarrow$ sparse)
- average degree 3.36
- average distance for the largest component: 7.64
- 6 degrees of separation [Travers & Milgram, 1967]
- heavy tail

Spanish flu in San Francisco 1918–1919

Small-world phenomenon and diseases

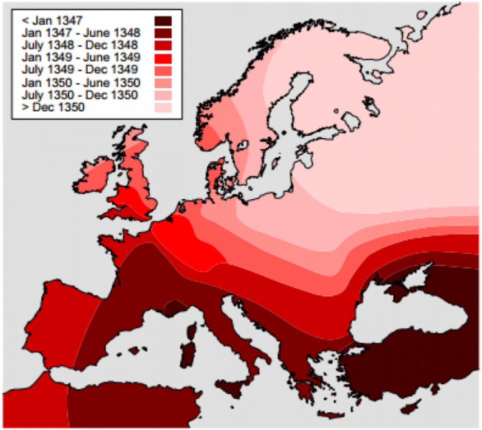


http://rsif.royalsocietypublishing.org/content/4/12/155

Small world: Obvious?

Black death!

Black death: spread



source: catholic.org

https://www.youtube.com/watch?v=EEK6c9Bh5CQ

Some of the other topics

- spectral graph theory, graph Laplacians, spectral clustering
- semi-supervised learning and manifold learning
- learnability on graphs - transductive learning
- online decision-making on graphs, graph bandits
- submodularity on graphs
- real-world graphs scalability and approximations
- spectral sparsification
- social network and recommender systems applications
- link prediction/link clasification
- signed networks (eOpinions)
- generalization bounds by perturbation analysis

Links to the other courses

- **Introduction to statistical learning**
  - links to the learning theory on graphs: label propagation, learnability, generalization
- **Reinforcement learning**
  - link to the online learning (bandit) lecture at the end of the semester
- **Advanced learning for text and graph data**
  - data-mining graph course on the topics not covered in this course
  - details on the next slide

MVA and Graphs: 2 courses

The two MVA graph courses offer complementary material.

Fall: **Graphs in ML**

this class

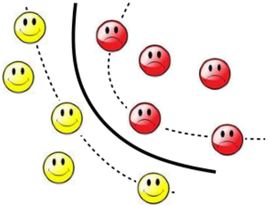
- focus on learning
- spectral clustering
- random walks
- graph Laplacian
- semi-supervised learning
- manifold learning
- theoretical analyses
- online learning
- recommender systems

Late fall: **ALTeGraD**

by Michalis Vazirgiannis

- dimensionality reduction
- feature selection
- text mining
- graph mining
- community mining
- graph generators
- graph-evaluation measures
- privacy in graph mining
- big data

Statistical Machine Learning in Paris!



https://sites.google.com/site/smileinparis/sessions-2016--17

Speakers: ML PhD students - former MVA students

Topic: ICML 2018 debrief

Date: Thursday October 4th

Time: 15:00 - 17:00 (this is pretty soon)

Place: Télécom ParisTech, Amphithéâtre Grenat

Administrivia

Time: Wednesdays afternoons, next week at 14:00

Place: ENS Cachan somewhere, next week at Salle Condorcet

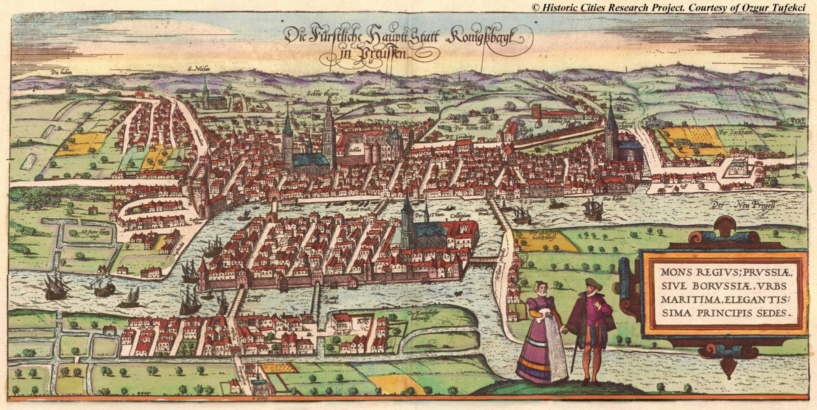
7 lectures: 3.10. 10.10. 16.10. 31.10. 7.11. 21.11. 12.12.
3 recitations (TDs): 24.10. 14.11. 28.11.

Validation: grades from TDs (40%) + class project (60%)
Research: contact me for *internships*, *PhD.theses*, *projects*, etc.

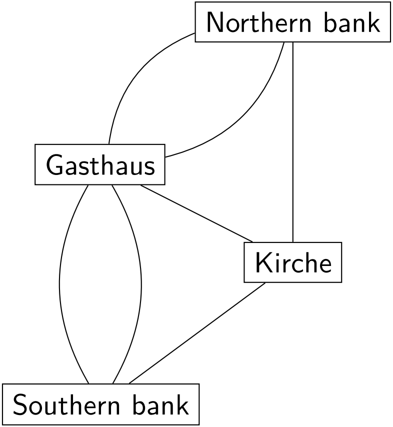
Course website:
`http://researchers.lille.inria.fr/~valko/hp/mva-ml-graphs`

Contact, online class discussions, and announcements:
`https://piazza.com/ens_cachan/fall2018/mvagraphsml`
class code given during the class

Graph theory refresher



Graph theory refresher



Graph theory refresher

- 250 years of graph theory
- Seven Bridges of Königsberg (Leonhard Euler, 1735)
- necessary for Eulerian circuit: 0 or 2 nodes of odd degree
- after bombing and rebuilding there are now 5 bridges in Kaliningrad for the nodes with degrees $[2, 2, 3, 3]$
- the original problem is solved but not practical
  http://people.engr.ncsu.edu/mfms/SevenBridges/

Similarity Graphs

Input: $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_N$

- raw data
- flat data
- vectorial data

Similarity Graphs

Similarity graph: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ — **(un)weighted**

Task 1: For each pair i, j: define a **similarity function** s_{ij}

Task 2: Decide which edges to include

ε-neighborhood graphs – connect the points with the distances smaller than ε

k-NN neighborhood graphs – take k nearest neighbors

fully connected graphs - consider everything

This is art (not much theory exists).

http://www.informatik.uni-hamburg.de/ML/contents/people/luxburg/publications/Luxburg07_tutorial.pdf

Similarity Graphs: ε-neighborhood graphs

Edges connect the points with the distances smaller than ε.

- distances are roughly on the same scale (ε)
- weights may not bring additional info $\rightarrow$ unweighted
- equivalent to: similarity function is at least ε
- theory [Penrose, 1999]: $\varepsilon = ((\log N)/N)^d$ to guarantee connectivity N nodes, d dimension
- practice: choose ε as the length of the longest edge in the MST - minimum spanning tree

What could be the problem with this MST approach?

Similarity Graphs: k-nearest neighbors graphs

Edges connect each node to its k-nearest neighbors.

- asymmetric (or directed graph)
  - option OR: ignore the direction
  - option AND: include if we have both direction (mutual k-NN)
- how to choose k?
- $k \approx \log N$ - suggested by asymptotics (practice: up to $\sqrt{N}$)
- for mutual k-NN we need to take larger k
- mutual k-NN does not connect regions with different density
- why don't we take $k = N - 1$?

Similarity Graphs: Fully connected graphs

Edges connect everything.

- choose a "meaningful" similarity function s
- default choice:

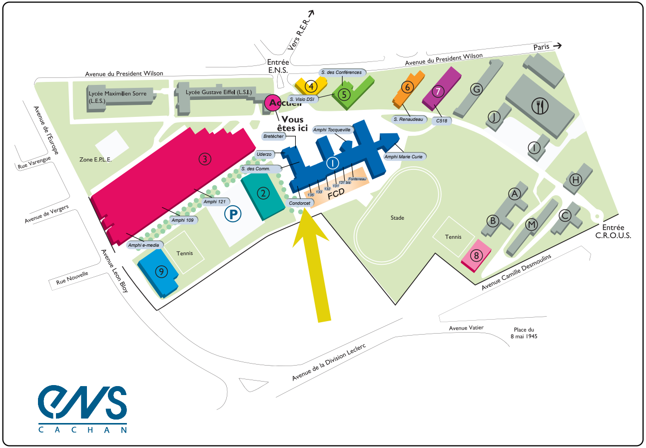
$$s_{ij} = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$$

- why the exponential decay with the distance?
- σ controls the width of the neighborhoods
  - similar role as ε
  - **a** practical rule of thumb: 10% of the average empirical std
  - possibility: learn σ_i for each feature independently
- metric learning (a whole field of ML)

Similarity Graphs: Important considerations

- *calculate all s_{ij} and threshold* has its limits ($N \approx 10000$)
- graph construction step can be a huge bottleneck
- want to go higher? (we often have to)
  - down-sample
  - approximate NN
    - **LSH** - Locally Sensitive Hashing
    - **CoverTrees**
    - **Spectral sparsifiers**
  - sometime we may not need the graph (just the final results)
  - yet another story: when we start with a large graph and want to make it sparse (later in the course)
- these rules have little theoretical underpinning
- similarity is very data-dependent

Next class on Wednesday, October 10th at 14:00!



Michal Valko
michal.valko@inria.fr

ENS Paris-Saclay, MVA 2018/2019

SequeL team, Inria Lille — Nord Europe
https://team.inria.fr/sequel/